

## REMARKS

Reconsideration of this application, as amended, is respectfully requested. Claim 59 has been amended to clarify that the conversion rules are for converting harvested content. This amendment is supported by the specification as filed; for example at paragraph [0059]. Accordingly, no new matter is added by this amendment.

- A. **Contrary to the conclusions set forth in the Office Action, claims 59 and 60 are patentable over Puder, “System Support for Knowledge-Based Trading in Open Service Markets”, which fails to teach or suggest storing rules for conversion of harvested content in a repository.**

As amended, claim 59 includes the feature of converting the harvested content based on conversion rules stored in the repository. Puder does not have any provision to convert such harvested content based on conversion rules stored in a repository. Instead, Puder describes a mediator-like system which matches together service requests and service offerings using conceptual graphs. (Puder, Abstract). The essential functions provided by this system are described as: service browsing, conceptual graph creation for type description, and service trading. (Puder, Section 1. Introduction and Section 2. Type Specification and Conceptual Graphs). Service trading requires the use of a service database which stores service types based on conceptual graphs, a lexicographical database that maintains the background knowledge needed to match two conceptual graphs, and a repository for matching rules which are used to define different metrics to compute the semantic distance between two graphs. (Puder, Section 3. Trading of Service Types). In particular, the rules stored in the repository are rules for matching and trading services in the open service market.

The “interceptor” described in Puder bridges border between two domain by converting “parameters of a function call according to the contra- and covariance rules”. (Puder, first two paragraphs under Figure 3 and Figure 3). In other words the rules used by interceptor are for converting function call parameters, i.e., basically for changing the parameter types so that, for example, a service request from a CORBA domain can be matched with service offered from a DCE domain.

This process of using rules to change function call parameters cannot be interpreted as converting harvested content based on conversion rules stored in a repository. Nowhere does Puder define rules for converting harvested content. For example, acquired content in Puder could be the result of a function call, and there are no conversion rules for the results of such function calls.

At best, Puder describes a scheme for storing rules for converting function call parameters, not rules for conversion of harvested. Consequently, for at least these reasons, Puder does not anticipate claims 59 and 60 and these rejections should be removed.

**B. Claim 61 is patentable over Puder and Kremen, which like Purder fails to teach or suggest storing rules for conversion of harvested content in a repository.**

Adding the teachings of Kremen, US Patent 5,706,434, to those of Puder does not alter the conclusions set forth above. Kremen describes an integrated request-response system that creates and serves data objects among various communication protocols by decoding incoming requests, identifying the protocol used for transmission, generating an abstract data object, merging data from a main database with the abstract data object, and then formatting the data object for outgoing transmission according to the protocol of the intended recipient. (Kremen Abstract). Such a scheme does not involve harvesting content from disparate sources nor converting the harvested content based on conversion rules stored in the repository, as recited in claim 59.

The repositories discussed by Kremen store data to be merged with objects and configuration information used to decode the requests and format the data objects. Kremen, col. 5 ll. 40-45. There is no mention of storing conversion rules to be used for harvested content and so the elements missing from Puder are likewise missing from Kremen. Consequently, claim 59 and its dependent claim 61 are patentable over this combination of references.

**C. Claims 1, 2, 6 and 7 are patentable over Ensink in view of Spencer because neither of these references teaches or suggests the creation and/or use of capture templates to harvest content from disparate sources.**

Ensink, “XML Based Adaptation of the Composite Approach for Database Integration”, discusses an approach for integrating data from heterogeneous databases in which outputs from customized programs run against various databases are made to conform to a selected XML template having standardized data descriptors (tags). See, Ensink, Section 5.1. In contrast, claim 1 recites a method in which capture templates are used to harvest content by controlling the extraction of data. Similar features are recited in claim 6. The output templates described by Ensink play no role in controlling any data extraction processes and, consequently, are significantly different from the capture templates as presently claimed.

The BookStore Markup Language (BSML) described in Ensink cannot be equated to the capture templates recited in the present claims because BSML just models a book; that is, the language defines how information regarding a book is to be represented. Ensink Section 5.1. BSML does not define how

information about a book can be harvested, say for example from the database that contains the information. In fact, Ensink states that “final step in data integration through XML is the creation of custom programs to generate the XML tagged data from databases. This data conforms to the standard template described in the object definition ... final step involved writing a specific program for each database to reproduce the data in BSML.” Ensink Section 5.1. Thus, specific programs are driving the extraction of data from databases, not BSML. For at least these reasons, Ensink cannot be viewed as teaching the use of capture templates to harvest content by controlling the extraction of data.

Adding the teachings of Spencer, “Using XML to Build Internet Solutions”, fails to cure these deficiencies. Spencer discusses standardization of data-exchange mechanisms using XML, Spencer Paragraph 1-2, and the benefits of using XML for standardized information delivery across the Internet. That is, Spencer extols the virtues of presenting data in a fashion that conforms to a particular output template (i.e., an XML-defined template) as a means of fostering easy presentations of such data. Spencer does not, however, teach or suggest the creation of capture templates that would control extraction of data. Hence, combining the teachings of Spencer with those of Ensink may allow for the presentation of the Ensink “bookstore” data in a standardized fashion but it still would not yield a scheme in which capture templates are used to harvest content by controlling the extraction of data as recited in claims 1 and 6. Therefore, for at least these reasons, claims 1 and 6, and their respective dependent claims, are patentable over the combination of Ensink and Spencer.

**D. The remaining dependent claims are patentable over Ensink and Spencer, even when considered in combination with Lonroth, Nassbaum and/or Arens.**

Claims 3 and 8 were rejected as being unpatentable over Ensink in view of Spencer and further in view of of Lonroth, U.S. Patent No. 6,826,597 and Nassbaum, U.S. Patent No. 6,779,154. Lonroth discusses a system and method for providing clients with services to retrieve data from data sources that do not necessarily support the protocol and format required by the clients. Lonroth, Abstract. This scheme does not involve the creation and use of capture templates to harvest content by extracting data under the control of the capture templates as recited in claims 1 and 6. Instead, intermediate response XML documents are created from received HTML content, those documents are filtered by selectively removing content according to filtering rules, and an XSL styling sheet is applied to format the response document according to another set of rules associated with the style sheet. Lonroth, Abstract. Neither the response XML document nor the XSL styling sheet described by Lonroth can be considered a capture template created to harvest content as recited in the present claims.

Nassbaum describes an application server that executes voice-enabled web applications by runtime execution of XML documents that define those applications. The application server includes an HTML conversion module configured for translating information present during runtime execution of an XML document into an HTML document. The system converts the XML document into an HTML document in a manner that is reversible, where all the information from the original XML document is preserved such that the HTML document can be converted back to the original XML document. The translation of the XML documents into HTML documents described by Nassbaum is quite distinct from the creation of capture templates to harvest content by extracting data under the control of the capture templates as recited in the present claims. For example, the HTML documents described by Nassbaum play no role in controlling any such data extraction.

Thus, adding the teachings of Lonnroth and/or Nassbaum to those of Ensink and Spencer would not alter the conclusions of patentability with respect to claims 1 and 6 set forth above. Because these independent claims would remain patentable over the combination of references it follows that dependent claims 3 and 8 would likewise be patentable over these references.

Claims 4 and 9 were rejected as being unpatentable over Ensink in view of Spencer and further in view of Lonnroth. Claim 4 depends from claim 1 and claim 9 depends from claim 6. Therefore, these claims are patentable for at least the same reasons as discussed above with respect to claims 3 and 8.

Claims 5 and 10 were rejected as being unpatentable over Ensink in view of Spencer and further in view of Arens, "Intelligent Caching: Selecting, Representing, and Reusing Data in an Information Server", which discusses caching results of queries and how to use such cached results for future queries. Arens, however, does not describe the creation and use of capture templates to harvest content by extracting data under the control of the capture templates as recited in independent claims 1 and 6 and the Office Action does not contend otherwise. Hence, the patentability of independent claims 1 and 6, and by implication their respective dependent claims 5 and 10, is not affected by adding the teachings of Arens. Stated differently, these claims remain patentable for at least the reasons set forth above.

For all of the foregoing reasons, the claims are patentable over the references cited in the Office Action. If there are any additional fees due in connection with this communication, please charge our deposit account no. 19-3140.

Respectfully submitted,

SONNENSCHN NATH & ROSENTHAL LLP

Dated: May 18, 2006

s/Tarek N. Fahmi/

Tarek N. Fahmi

Reg. No. 41,402

P.O. Box 061080  
Wacker drive Station  
Sears Tower  
Chicago, IL 60606-1080  
(415) 882-5023